

CAPSTONE PROJECT

ONLINE QUIZ APPLICATION

Presented By:

1. **Name** : B Surya Teja
2. **College Name** : Presidency University
3. **Department** : Computer Science Engineering

OUTLINE

- Problem Statement
- System Development Approach
- Algorithm & Deployment
- Result
- Conclusion
- Future Scope
- References

PROBLEM STATEMENT

Traditional quiz applications often lack flexibility, personalization, and offline accessibility. Most systems rely only on fixed question sets without supporting custom question management. Users face limited control over quiz settings like timer, difficulty, or infinite mode. Admins usually struggle to manage large question banks efficiently in a lightweight system. There is also a need for history tracking and performance analysis for learners. Hence, a smart, user-friendly, browser-based quiz system is required to overcome these limitations.

SYSTEM APPROACH

□ Frontend Technologies

- **HTML5:** For semantic page structure and panels.
- **CSS3 / Flexbox / Grid:** For responsive layout, modern styling, gradient backgrounds, animations, and custom components.
- **JavaScript (ES6): Core logic for:**
 - Authentication
 - Quiz management and timer
 - Local Storage persistence
 - Admin panel CRUD operations
 - Dynamic DOM manipulation and event handling
- **Fonts:** Google Fonts-Poppins.
- **External API:** Open Trivia DB (fetch fallback questions).

SYSTEM APPROACH

❑ Design Approach

- **User-Centered Design:**
 - Clear separation of login/signup, quiz setup, quiz panel, and results.
 - Immediate feedback using overlay messages for correct/wrong answers.
 - Sidebar shows live score, progress, and quick actions for easy access.
- **Responsive & Interactive Layout:**
 - CSS Grid and Flexbox for main layout and sidebar.
 - Hover and click animations for buttons and quiz choices.
 - Timer bar with dynamic width animation.
- **Colour & Theme:**
 - Dark theme with gradient accents.
 - Success/Failure colours for quiz feedback.
 - Consistent visual hierarchy for readability and user experience.
- **SPA Behaviour:**
 - Panels shown/hidden dynamically without page reloads.
 - Smooth quiz progression with randomized question order and infinite mode support.

ALGORITHM & DEPLOYMENT

- **Product Catalog Creation**

- Define the quiz structure (qualification, subject, course, and language).
- Store questions in a structured format within local Storage as a catalog.
- Provide admin access to add, edit, and delete questions dynamically.

- **Frontend Design**

- Use HTML5, CSS3 (Grid/Flexbox), and Google Fonts for responsive layout.
- Implement dark-themed UI with gradient accents for better engagement.
- Organize panels for login, quiz setup, quiz runtime, and admin management.

- **Interactive Features**

- Enable timers, progress tracking, overlays for correct/wrong answers.
- Support infinite quizzes, customizable settings, and performance history.
- Add admin tools for CRUD operations on question banks with live preview.

- **Testing**

- Perform unit testing on quiz logic (timers, scoring, randomization).
- Conduct UI/UX testing for responsiveness across devices.
- Validate local Storage persistence and Open Trivia DB integration.

PROJECT CODE

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8" />
5 <meta name="viewport" content="width=device-width,initial-scale=1" />
6 <title>Smart Quiz - Admin + OpenTrivia</title>
7 <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600;700&display=swap" rel="stylesheet">
8 <style>
9
10 :root{
11 --rad: 10px; --rad2: 15px; --rad3: 20px; --text: #fff; --muted: #999;
12 --accent: linear-gradient(90deg,var(--rad),var(--rad2),var(--rad3));
13 --success: #006666; --danger: #ff5252;
14 }
15
16 *{
17 box-sizing: border-box;
18 }
19
20 body{
21 margin: 0;
22 font-family: 'Poppins', system-ui, Arial, sans-serif;
23 background: linear-gradient(135deg,var(--rad),var(--rad2),var(--rad3));
24 background-size: 400% 400%; animation: move 18s linear infinite;
25 min-height: 100vh; display: flex; align-items: center; justify-content: center; color: var(--text);
26 }
27
28 @keyframes move{
29 0%{background-position: 0% 50%;}
30 50%{background-position: 100% 50%;}
31 100%{background-position: 0% 50%;}
32 }
33
34 .app{
35 width: 100%; max-width: 1100px; margin: 10px; background: var(--card); border-radius: 10px; box-shadow: 0 0 0 1px #000; overflow: hidden;
36 }
37
38 header{
39 display: flex; align-items: center; justify-content: space-between; padding: 10px 20px; background: linear-gradient(90deg, #006666, #006666); color: #fff;
40 }
41
42 .brand{
43 display: flex; align-items: center; gap: 10px;
44 }
45
46 .logo{
47 width: 40px; height: 40px; border-radius: 10px; background: var(--accent); display: flex; align-items: center; justify-content: center; font-weight: 700; color: #fff; box-shadow: 0 0 0 1px #000;
48 }
49
50 .title{
51 font-size: 1.2em; font-weight: 700;
52 }
53
54 .user-area{
55 display: flex; align-items: center; gap: 10px;
56 }
57
58 .btn{
59 padding: 5px 10px; border-radius: 10px; font-weight: 600; cursor: pointer; border: 0; transition: all .1s; background: #006666; color: var(--text);
60 }
61
62 .btn:hover{
63 transform: translateY(-4px); box-shadow: 0 10px 20px #006666;
64 }
65
66 .btn.primary{
67 background: var(--accent); color: #fff;
68 }
69
70 .btn.ghost{
71 background: transparent; border: 1px solid #006666; color: var(--text);
72 }
73
74 .main{
75 display: grid; grid-template-columns: 1fr 300px; gap: 10px; padding: 10px;
76 }
77
78 @media(max-width: 1000px){
79 .main{
80 grid-template-columns: 1fr;
81 }
82 }
83
84 .panel{
85 background: var(--card); padding: 10px; border-radius: 10px; box-shadow: inset 0 1px 0 #006666;
86 }
87
88 h1.page-title{
89 font-size: 1.2em; margin: 0 10px; font-weight: 700;
90 }
91
92 label{
93 font-size: 0.9em; color: var(--muted); font-weight: 600; margin: 5px 0 5px 10px; font-size: 0.9em;
94 }
95
96 select,input,input[type="text"],input{
97 width: 100%; padding: 10px; border-radius: 10px; border: 0; background: #fff; color: #000; outline: none; font-size: 0.9em; transition: all .1s;
98 }
99
100 .checkbox{
101 width: 15px; height: 15px; border-radius: 50%; border: 1px solid #006666; display: flex; align-items: center; justify-content: center; margin: 5px 0 5px 10px; font-size: 0.8em;
102 }
```

```
1 <html lang="en">
2
3 <head>
4 <style>
5
6 select,input option{
7 color: #000; background: #fff;
8 }
9
10 select,input:focus,input[type="text"],input[type="password"],input[type="text"]:focus{
11 box-shadow: 0 0 0 1px #006666;
12 }
13
14 .small{
15 font-size: 0.8em; color: var(--muted);
16 }
17
18 .row{
19 display: flex; gap: 10px; align-items: center;
20 }
21
22 .row.two{
23 flex: 1;
24 }
25
26 .actions{
27 display: flex; gap: 10px; margin-top: 10px; flex-wrap: wrap;
28 }
29
30 .choice{
31 padding: 10px; border-radius: 10px; background: #006666; color: #fff; cursor: pointer; border: 1px solid #006666; transition: all .1s; color: var(--text);
32 }
33
34 .choice:hover{
35 transform: translateY(-4px); box-shadow: 0 10px 20px #006666;
36 }
37
38 .choice.correct{
39 background: #006666; border-radius: 10px; border: 1px solid #006666;
40 }
41
42 .choice.wrong{
43 background: #ff5252; border-radius: 10px; border: 1px solid #ff5252;
44 }
45
46 .timerbar{
47 height: 10px; background: #006666; border-radius: 10px; width: 100%;
48 }
49
50 .timerfill{
51 height: 10px; background: linear-gradient(90deg,var(--rad),var(--rad2),var(--rad3)); width: 100%; transition: width 1s linear;
52 }
53
54 .sidebar{
55 display: flex; flex-direction: column; gap: 10px;
56 }
57
58 .big{
59 font-size: 1.2em; font-weight: 700;
60 }
61
62 .history-list{
63 max-height: 150px; overflow: auto; padding-right: 10px;
64 }
65
66 .history-item{
67 padding: 10px; border-radius: 10px; background: linear-gradient(90deg, #006666, #006666); margin-bottom: 10px; border: 1px solid #006666;
68 }
69
70 .label{
71 color: var(--muted); font-size: 0.8em;
72 }
73
74 .overlay-msg{
75 position: fixed; left: 50%; top: 50%; transform: translate(-50%, -50%); background: #006666; padding: 20px 20px; border-radius: 10px; color: #fff; font-size: 1.2em;
76 }
77
78 /* History modal body lg */
79
80 .historyModal .hist-body{
81 background: linear-gradient(135deg, #006666, #006666); color: #fff; padding: 10px; border-radius: 10px;
82 }
83
84 /* Admin modal items */
85
86 .admin-list{
87 max-height: 200px; overflow: auto; margin-top: 10px;
88 }
89
90 .admin-item{
91 padding: 10px; background: #006666; margin-bottom: 10px; display: flex; gap: 10px; align-items: center; justify-content: space-between; border-radius: 10px;
92 }
93
94 footer{
95 padding: 10px 10px; text-align: center; color: var(--muted); font-size: 0.8em;
96 }
97
98 </style>
99
100 </head>
101
102 <body>
103
104 <div class="app">
105 <header>
106 <div class="brand">
107 <div class="logo"><img alt="Logo" /></div>
108 </div>
109 </header>
110
111 <div class="main">
112 <div class="panel">
113 <div class="title">Smart Quiz Application</div>
114 <div class="small">Admin-built questions + Open Trivia DB fallback</div>
115 </div>
116
117 <div class="user-area" id="userArea">
118 <div id="welcomeGuest" style="display: flex; gap: 10px; align-items: center;">
119 <button class="btn" onclick="showTab('login')">Login</button>
120 <button class="btn" onclick="showTab('signup')">Sign Up</button>
121 <button class="btn" onclick="showTab('adminAuth')">Admin Login / Signup</button>
122 </div>
123
124 <div id="loggedInArea" style="display: none; gap: 10px; align-items: center;">
125 <div id="loggedUser" style="padding: 5px 10px; background: #006666; color: #fff; border-radius: 10px; border: 1px solid #006666;"></div>
126 <button class="btn" onclick="openHistory()">History</button>
127 <button class="btn" id="adminPanelBtn" onclick="openAdminPanel()" style="display: none">Admin Panel</button>
128 <button class="btn" onclick="logout()">Logout</button>
129 </div>
130 </div>
131
132 </header>
133
134 <div class="main">
135 <div>
136 <!-- LOGIN TAB -->
137 <div class="panel" id="tab-login">
138 <div class="page-title">Login</div>
139 <label class="row-label">Username</label>
140 <input id="loginUsername" class="input" type="text" placeholder="username">
141 <label class="row-label">Password</label>
142 <input id="loginPassword" class="input" type="password" placeholder="password">
143 <div class="actions">
144 <button class="btn primary" onclick="userLogin()">Login</button>
145 <button class="btn ghost" onclick="showTab('signup')">Go to Signup</button>
146 </div>
147 </div>
148
149 <!-- SIGNUP TAB -->
150 <div class="panel" id="tab-signup" style="display: none">
151 </div>
152 </div>
153 </div>
154 </div>
155 </body>
156 </html>
```

```
1 <div class="panel" id="tab-login">
2 <div class="page-title">Login</div>
3 <label class="row-label">Username</label>
4 <input id="loginUsername" class="input" type="text" placeholder="username">
5 <label class="row-label">Password</label>
6 <input id="loginPassword" class="input" type="password" placeholder="password">
7 <div class="actions">
8 <button class="btn primary" onclick="userLogin()">Login</button>
9 <button class="btn ghost" onclick="showTab('signup')">Go to Signup</button>
10 </div>
11 </div>
12
13 <!-- SIGNUP TAB -->
14 <div class="panel" id="tab-signup" style="display: none">
15 </div>
16 </div>
17 </div>
18 </body>
19 </html>
```

PROJECT CODE

The image shows a web browser window with a task manager interface. The browser's address bar displays the URL "http://localhost:3000/task_manager.html". The page title is "Task Manager". The main content area shows a quiz panel with the following elements: a title "Quiz", a question "What is the capital of France?", a timer "Time left: 10:00", a progress bar, and a list of choices: "Paris", "London", "Berlin", and "Rome". The "Paris" choice is selected. Below the choices is a "Stop Quiz" button. The browser's developer tools are open, showing the HTML structure of the page. The HTML includes a header with a title and a main container with a quiz panel. The quiz panel contains a title, a question, a timer, a progress bar, a list of choices, and a "Stop Quiz" button. The HTML is rendered in a light blue theme.

The screenshot shows a web browser window displaying a "Task Manager" application. The address bar indicates the URL is `http://localhost:3000/task-manager.html`. The page title is "Task Manager". The main content area displays a form with fields for "Username" and "Password", and a "Login" button. Below the form, there are links for "Sign Up" and "Forgot Password". A sidebar on the right contains navigation links: "Home", "Users", "Groups", "Roles", "Permissions", "Audit Log", "Settings", and "Help".

The browser's developer tools are open, showing the "Console" tab with a red error message:

```
Error: Uncaught SyntaxError: Unexpected token ':'  
    at <anonymous>:278:1  
    at <anonymous>:278:1
```

The source code editor shows the following JavaScript code snippet around line 278:

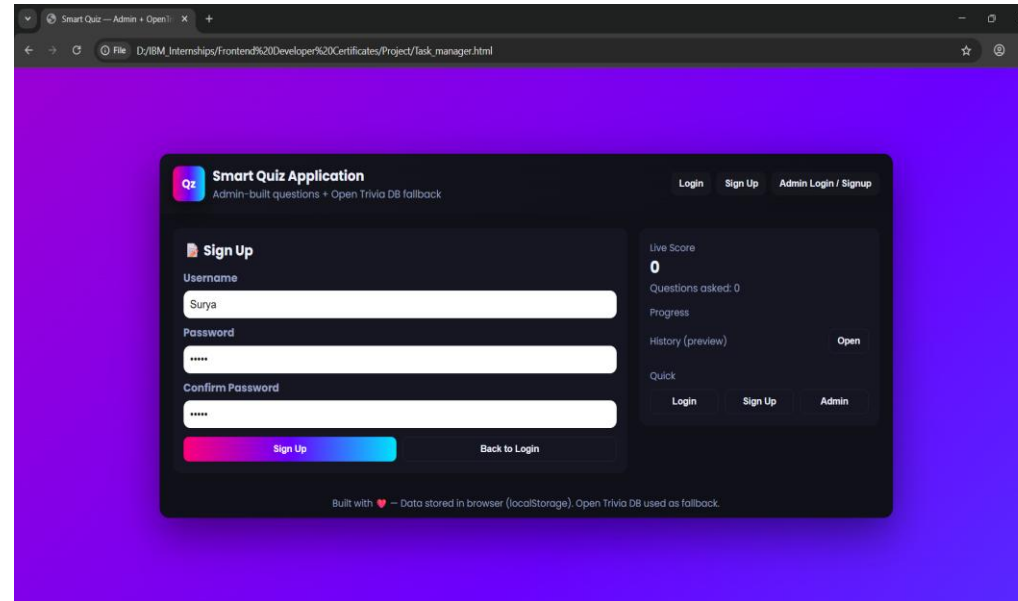
```
<script>  
  // Storage keys & initial state  
  const STORAGE_USERS = 'smartquiz_users_v2';  
  const STORAGE_ADMINS = 'smartquiz_admins_v1';  
  const STORAGE_QBANK = 'smartquiz_qbank_v1';  
  
  // default admin fallback (only used if no admins exist in storage)  
  const DEFAULT_ADMIN = { username: 'admin', password: 'admin123' };  
  
  let currentUser = null;  
  let isAdminSession = false;  
  
  /* UI helpers */  
  function showOverlay(msg, ms=1500){ const el=document.getElementById('overlayMsg'); el.textContent = msg; el.style.display='block'; setTimeout(()=>el.style.display='none', ms)  
  function showTab(tab){  
    // tabs: login, signup, adminauth  
    document.getElementById('tab-login').style.display = (tab==='login') ? 'block' : 'none';  
    document.getElementById('tab-signup').style.display = (tab==='signup') ? 'block' : 'none';  
    document.getElementById('tab-adminauth').style.display = (tab==='adminauth') ? 'block' : 'none';  
  
    function hideAllMainPanels(){  
      ['setupPanel','quizPanel','endPanel'].forEach(id=>document.getElementById(id).style.display='none');  
    }  
  
    /* ----- AUTH: Users ----- */  
    function userSignup(){  
      const u = document.getElementById('signupUsername').value.trim();
```

```

710 modal.querySelector("#adminCloseBtn").onclick = () => modal.remove();
711
712 // Add question
713 modal.querySelector("#addAddBtn").onclick = () => {
714   const qual = admsub.value; const subj = admsub.value; const course = (modal.querySelector("#admcourseKey").style.display === 'block') ? admsub.value : null;
715   const qtxt = modal.querySelector("#addq").value.trim(); const corr = modal.querySelector("#addCorrect").value.trim();
716   const i1 = modal.querySelector("#addinc1").value.trim(); const i2 = modal.querySelector("#addinc2").value.trim(); const i3 = modal.querySelector("#addinc3").value.trim();
717   if (qual & i1 & i2 & i3 & qtxt & corr) { showOverlay('Please fill mandatory fields', return; }
718   const obj = { e: qtxt, correct: corr, incorrect: [i1, i2, i3] }.filter(Boolean);
719   addQuestionToBank(qual, subj, course, obj);
720   showOverlay('Question added');
721   modal.querySelector("#addq").value = ''; modal.querySelector("#admcourseKey").value = ''; modal.querySelector("#addinc1").value = ''; modal.querySelector("#addinc2").value = ''; modal
722   refreshAdminList();
723 };
724
725 // Edit / delete delegate
726 let editingRef = null;
727 modal.querySelector("#adminList").addEventListener('click', (e) => {
728   const target = e.target;
729   if (target.matches('.adm-edit')) {
730     const ref = JSON.parse(target.getAttribute('data-ref'));
731     const { qualKey, subKey, courseKey, idx } = ref;
732     editingRef = ref;
733     // load item into fields
734     admsub.value = qualKey; admsub.onChange();
735     setTimeout(() => admsub.value = subKey; admsub.onChange(); if (courseKey) setTimeout(() => admsub.value = courseKey; }, 100); }, 100);
736     const item = courseKey ? qbank[qualKey][subKey][courseKey][idx] : qbank[qualKey][subKey][idx];
737     modal.querySelector("#addq").value = item.q; modal.querySelector("#addCorrect").value = item.c; modal.querySelector("#addinc1").value = (item.incorrect & item
738     modal.querySelector("#addAddBtn").style.display = 'none'; modal.querySelector("#addDeleteBtn").style.display = 'inline-block';
739   } else if (target.matches('.adm-del')) {
740     const ref = JSON.parse(target.getAttribute('data-ref'));
741     if (!confirm('Delete this question?')) return;
742     const { qualKey, subKey, courseKey, idx } = ref;
743     if (courseKey) { qbank[qualKey][subKey][courseKey].splice(idx, 1); if (qbank[qualKey][subKey][courseKey].length === 0) delete qbank[qualKey][subKey][courseKey]; }
744     else { qbank[qualKey][subKey].splice(idx, 1); if (qbank[qualKey][subKey].length === 0) delete qbank[qualKey][subKey]; }
745     saveBank(); refreshAdminList(); showOverlay('Deleted');
746   }
747 });
748

```


RESULT



Smart Quiz Application
Admin-built questions + Open Trivia DB fallback

Sign Up

Username
Surya

Password

Confirm Password

Sign Up Back to Login

Live Score
0

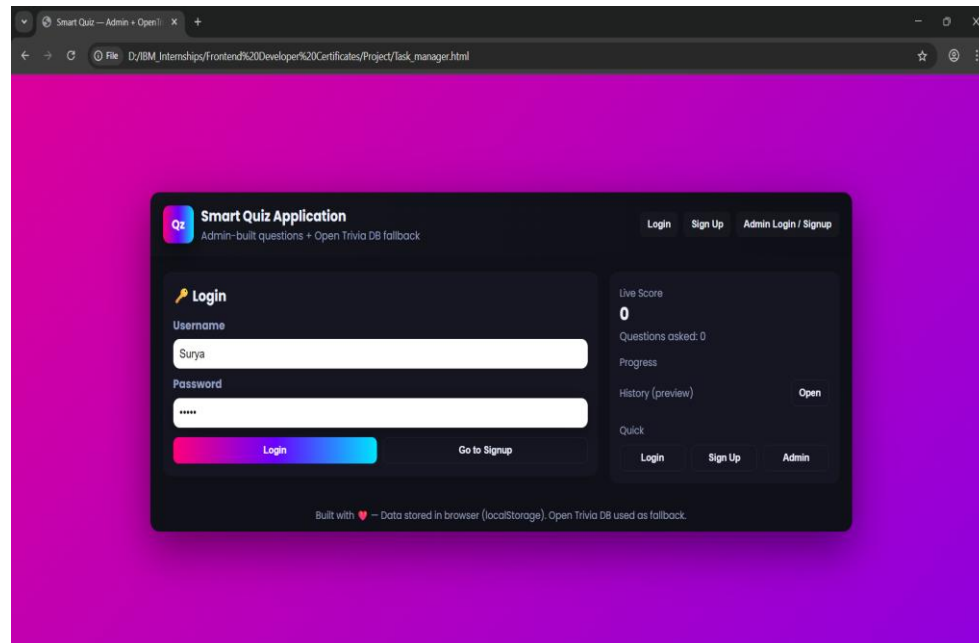
Questions asked: 0

Progress

History (preview) Open

Quick
Login Sign Up Admin

Built with ❤️ - Data stored in browser (localStorage). Open Trivia DB used as fallback.



Smart Quiz Application
Admin-built questions + Open Trivia DB fallback

Login

Username
Surya

Password

Login Go to Signup

Live Score
0

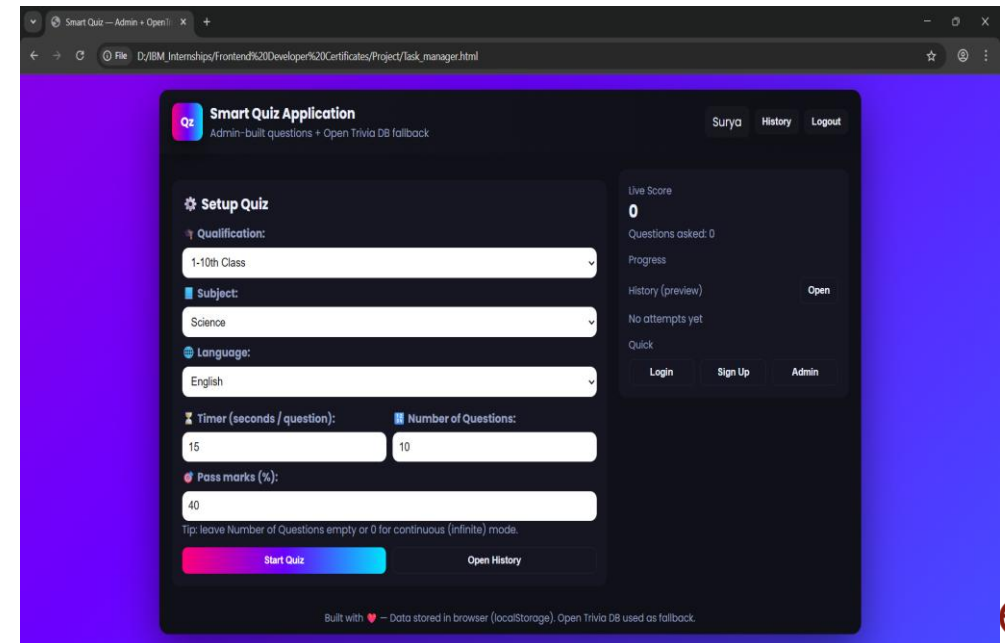
Questions asked: 0

Progress

History (preview) Open

Quick
Login Sign Up Admin

Built with ❤️ - Data stored in browser (localStorage). Open Trivia DB used as fallback.



Smart Quiz Application
Admin-built questions + Open Trivia DB fallback

Setup Quiz

Qualification:
1-10th Class

Subject:
Science

Language:
English

Timer (seconds / question):
15

Number of Questions:
10

Pass marks (%):
40

Tip: leave Number of Questions empty or 0 for continuous (infinite) mode.

Start Quiz Open History

Live Score
0

Questions asked: 0

Progress

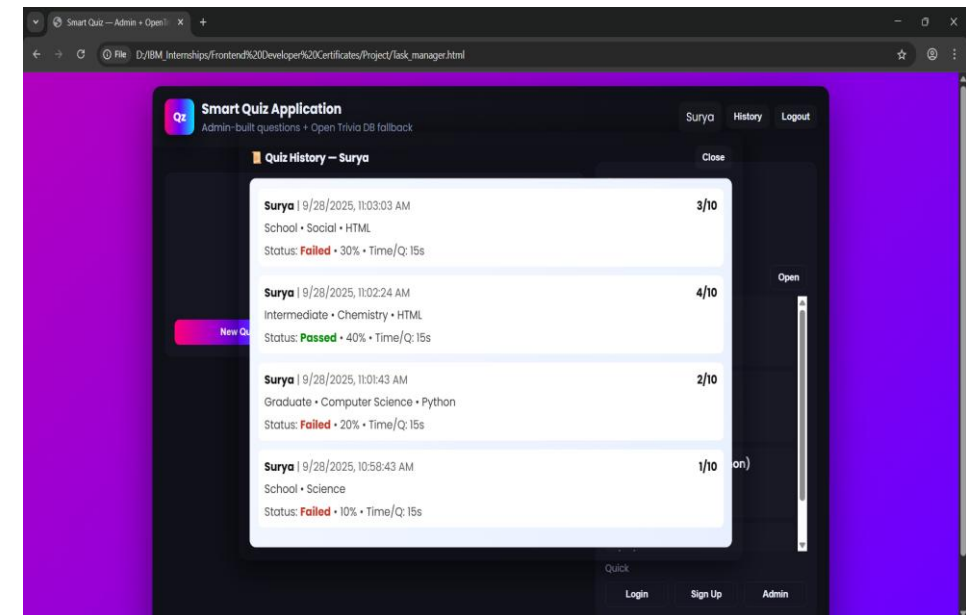
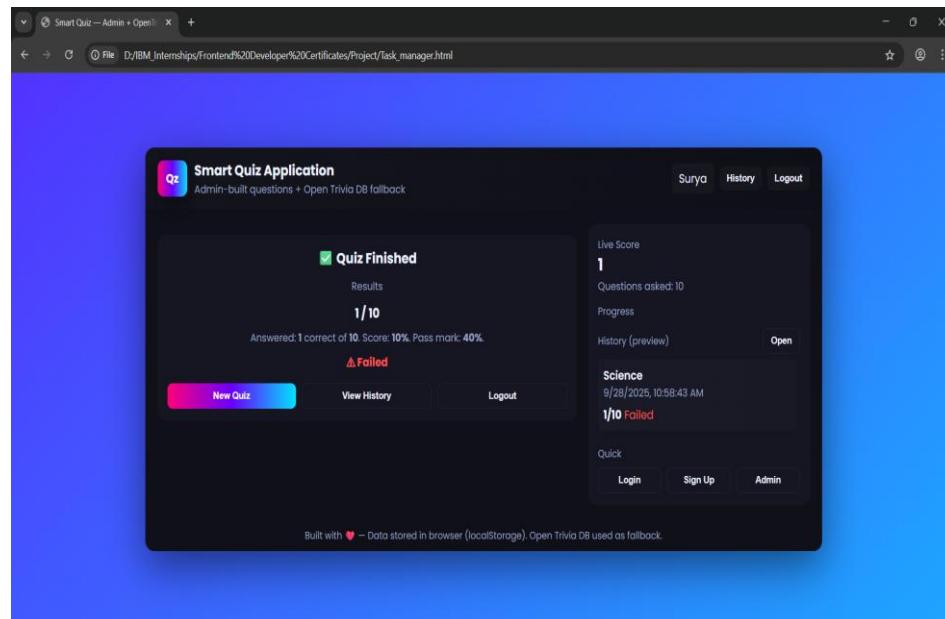
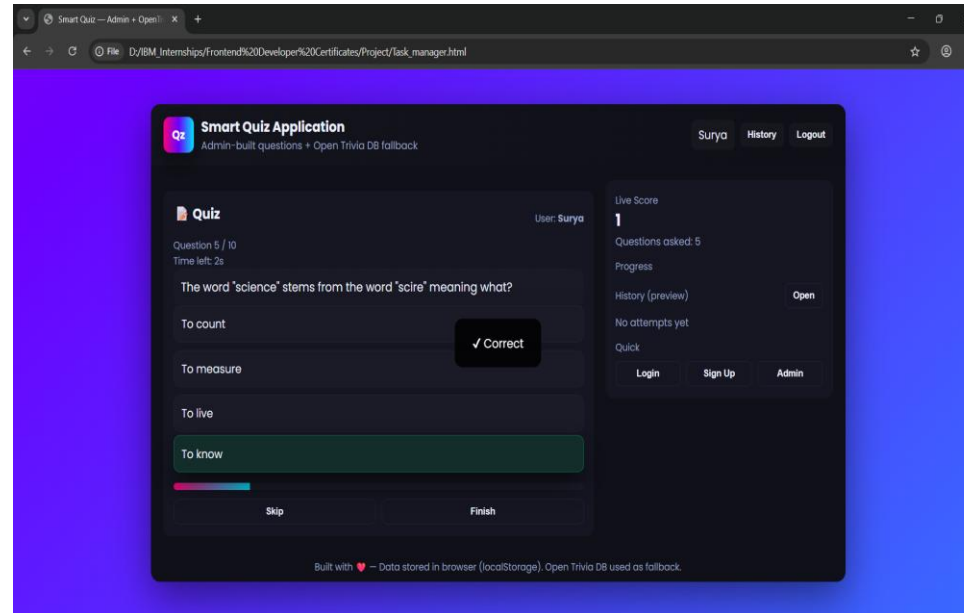
History (preview) Open

No attempts yet

Quick
Login Sign Up Admin

Built with ❤️ - Data stored in browser (localStorage). Open Trivia DB used as fallback.

RESULT



GITHUB AND DEPLOYMNET LINK

- **Attach your Github Link :** <https://github.com/SuryaTeja122/Online-Quiz-Application-Internship-Project.git>
- **Deployment link:** <https://suryateja122.github.io/Online-Quiz-Application-Internship-Project/>

CONCLUSION

The Online Quiz Application demonstrates an effective integration of frontend technologies and interactive design to provide a seamless user experience. Through systematic development from product catalog creation, frontend design, to interactive feature implementation the system ensures accuracy, usability, and engagement for users. Testing and deployment further validate the application's reliability and performance. Overall, this project highlights the importance of structured development approaches and modern technologies in creating practical, user-friendly educational tools.

FUTURE SCOPE(OPTIONAL)

- **Mobile Application Development:** The quiz system can be extended to mobile platforms (Android/iOS) to enhance accessibility.
- **AI-Based Question Generation:** Implement AI algorithms to automatically generate and categorize questions based on difficulty levels.
- **Analytics and Reporting:** Integrate advanced analytics to track user performance, strengths, and weaknesses for personalized learning.
- **Gamification Features:** Add leaderboards, badges, and rewards to increase user engagement and motivation.
- **Multilingual Support:** Expand the application to support multiple languages for a wider audience.
- **Integration with Learning Platforms:** Connect with e-learning portals or educational institutes for seamless content management.

REFERENCES

- <https://seldomindia.com/building-a-simple-quiz-app-with-html-css-and-javascript/>
- <https://www.scribd.com/document/579414230/Mini-Project-Report-quiz>
- <https://dev.to/linusmwiti21/level-up-your-skills-build-a-web-quiz-app-with-basic-js-html-css-lm5>
- <https://ijarsct.co.in/Paper24623.pdf>



THANK YOU